



软件职业技术学院“十一五”规划教材

Visual Basic

程序设计与应用

曾强聪 主 编

唐伟奇 赵 歆 龚中良 刘 震 副主编



中国水利水电出版社
www.waterpub.com.cn

软件职业技术学院“十一五”规划教材

Visual Basic 程序设计与应用

曾强聪 主 编

唐伟奇 赵 歆 龚中良 刘 震 副主编

邓顺川 主 审

中国水利水电出版社

内 容 提 要

本书是一本面向高职高专或应用型本科的 Visual Basic 教材。全书共 9 章, 涉及 Visual Basic 概述、数据类型、程序结构、可视化编程、界面设计、文件处理、图形处理、多媒体应用、网络应用、数据库应用、Active X 部件开发等多方面内容。考虑到高职高专与应用型本科 Visual Basic 教学的基础性与应用性, 本书采用了过程教学与实例教学相结合的编排结构, 前 5 章是 Visual Basic 基本特征、基本编程方法、基础程序结构的介绍, 后 4 章则是在已有基础学习上的基于示例程序的实际编程应用。

本书中有大量的示例程序, 这些程序大多取材于实际应用, 具有良好的编码规范, 并都根据教学需要进行了合理压缩或重新设计, 因此本书具有很强的实用性与可操作性。

本书可作为对 Visual Basic 有一定应用要求的计算机类、信息类专业的教材, 对于其他 Visual Basic 编程学习者或有关编程技术人员, 本书也有一定的参考价值。本书每章都配有习题与操作题, 并向读者提供书中全部的源程序代码、与教材配套的电子教案(用 PowerPoint 制作)、习题答案等。

本书电子教案和书中实例的源代码可以从中国水利水电出版社网站上免费下载, 网址为: <http://www.waterpub.com.cn/softdown/>。

图书在版编目(CIP)数据

Visual Basic 程序设计与应用 / 曾强聪主编. —北京:

中国水利水电出版社, 2007

软件职业技术学院“十一五”规划教材

ISBN 978-7-5084-4918-0

I. V… II. 曾… III. BASIC 语言—程序设计—高等学校:
技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 131423 号

书 名	Visual Basic 程序设计与应用
主 编	曾强聪
副 主 编	唐伟奇 赵 歆 龚中良 刘 震
主 审	邓顺川
出版 发行	中国水利水电出版社(北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787mm×1092mm 16 开本 19.25 印张 468 千字
版 次	2007 年 9 月第 1 版 2007 年 9 月第 1 次印刷
印 数	0001—4000 册
定 价	29.00 元

凡购买我社图书, 如有缺页、倒页、脱页的, 本社营销中心负责调换

版权所有·侵权必究

序

随着信息技术的广泛应用和互联网的迅猛发展,以信息产业发展水平为主要特征的综合国力竞争日趋激烈,软件产业作为信息产业的核心和国民经济信息化的基础,越来越受到世界各国的高度重视。中国加入世贸组织后,必须以积极的姿态,在更大范围和更深程度上参与国际合作和竞争。在这种形势下,摆在我们面前的突出问题是人才短缺,计算机应用与软件技术专业领域技能型人才的缺乏尤为突出,无论是数量还是质量,都远不能适应国内软件产业的发展和信息化建设的需要。因此,深化教育教学改革,推动高等职业教育与培训的全面发展,大力提高教学质量,是迫在眉睫的重要任务。

2000年6月,国务院发布《鼓励软件产业和集成电路产业发展的若干政策》,明确提出鼓励资金、人才等资源投向软件产业,并要求教育部门根据市场需求进一步扩大软件人才培养规模,依托高等学校、科研院所,建立一批软件人才培养基地。2002年9月,国务院办公厅转发了国务院信息化工作办公室制定的《振兴软件产业行动纲要》,该《纲要》明确提出要改善软件人才结构,大规模培养软件初级编程人员,满足软件工业化生产的需要。教育部也于2001年12月在35所大学启动了示范性软件学院的建设工作,并于2003年11月启动了试办示范性软件职业技术学院的建设工作。

示范性软件职业技术学院的建设目标是:经过几年努力,建设一批能够培养大量具有竞争能力的实用型软件职业技术人才的基地,面向就业、产学结合,为我国专科层次软件职业技术人才培养起到示范作用,并以此推动高等职业技术教育人才培养体系与管理体制和运行机制的改革。要达到这个目标,建立一套适合软件职业技术学院人才培养模式的教材体系显得尤为重要。

高职高专的教材建设已经走过了几个发展阶段,由最开始本科教材的压缩到加大实践性教学环节的比重,再到强调实践性教学环节,但是学生在学习时还是反映存在理论与实践的结合问题。为此,中国水利水电出版社在经过深入调查研究后,组织了一批长期工作在高职高专教学一线的老师,编写了这套“软件职业技术学院‘十一五’规划教材”,本套教材采用项目驱动的方法来编写,即全书所有章节都以实例作引导来说明各知识点,各章实例之间并不是孤立的,每个实例都可以作为最终项目的一个组成部分;每一章章末还配有实习实训(或叫实验),这些实训组合起来是一个完整的项目。

采用这种方式编写的图书与市场上同类教材相比更具优越性,学生不仅仅学到了知识点,还通过项目将这些知识点连成一条线,开拓了思路,掌握了知识,达到了面向岗位的职业教育培训目标。

本套教材的主要特点有:

- (1) 课程主辅分明——重点突出,教学内容实用。
- (2) 内容衔接合理——完全按项目运作所需的知识体系结构设置。

(3) 突出实习实训——重在培养学生的专业能力和实践能力,力求缩短人才与企业间的磨合期。

(4) 教材配套齐全——本套教材不仅包括教学用书,还包括实习实训材料、教学课件等,使用方便。

本套教材适用于广大计算机专业和非计算机专业的大中专院校的学生学习,也可作为有志于学习计算机软件技术与开发的工程技术人员的参考教材。

编委会

2006 年 7 月

前 言

Visual Basic 6.0 是一个集工程管理、界面设计、编码编译于一体的软件集成开发环境, 具有可视化、面向对象、部件开发等一系列特性, 能够有效地提高应用软件的开发效率。Visual Basic 6.0 还容易学习与使用, 对硬件配置要求低, 但功能强大、资源丰富、适应性好, 因此成为具有最广泛用户的软件开发平台。

最近几年里, Visual Basic 6.0 还成为了诸多高校, 尤其是高职高专计算机相关专业应用编程学习最主要的环境。本书即为高职高专计算机相关专业 Visual Basic 6.0 应用编程教学编写。本书本着科学严谨、结构合理、概念清晰的原则, 对 Visual Basic 6.0 进行较全面的介绍。全书共 9 章内容, 包括: Visual Basic 概述、数据类型、程序结构、可视化编程、界面设计、文件处理、图形处理、多媒体应用、网络应用、数据库应用、ActiveX 部件开发等多方面内容。

本书在内容组织与编排上充分考虑了高职高专计算机编程教学的基础性与应用性, 采用了过程教学与实例教学相结合的教学方法。前 5 章着重介绍 Visual Basic 的基本特征、基本编程方法、基础程序结构等; 后 4 章则着重在已有学习基础上的实际编程应用, 具有了很强的实用性与可操作性。

书中有大量示例程序, 大多取材于实际应用, 具有良好的编码规范, 并都根据教学需要进行了合理压缩或重新设计, 能更好地适应课堂教学, 是很好的学习范本。本书每章都有小结、习题, 可方便读者进行复习总结与练习。每章还配有操作题, 可方便实训教学的开展。为了方便教学与读者自学, 本书将提供全部源程序代码, 并提供与教材配套的电子教案(用 PowerPoint 制作)、习题答案等, 以上内容都可以从出版社网站下载。

本书由曾强聪任主编, 唐伟奇、赵歆、龚中良、刘震任副主编, 邓顺川主审, 周纳、丁启华、罗毅辉、王雷、陈英、孟洋、陈翠娥、曾义聪、曹伟、李波等参加了审稿讨论与教学试用。本书是湖南省教育科学“十一五”规划课题《高等职业教育中软件工程实用人才培养新模式的研究与实践》(课题号: ZC062) 的相关研究内容之一, 编写过程中得到了湖南省教育科学研究院的大力支持, 在此特表示感谢!

由于时间仓促, 作者水平有限, 书中难免有不足和疏忽之处, 恳请广大读者批评指正。

编 者

2007 年 6 月

目 录

序

前言

第 1 章 Visual Basic 概述.....	1
1.1 Visual Basic 的历史渊源	1
1.1.1 早期 BASIC.....	1
1.1.2 Quick BASIC.....	1
1.1.3 Visual Basic	2
1.2 Visual Basic 6.0 的特点	3
1.2.1 可视化	3
1.2.2 面向对象	3
1.2.3 部件技术	3
1.2.4 事件驱动	4
1.3 安装 Visual Basic 6.0	4
1.3.1 Visual Basic 6.0 对环境的要求	4
1.3.2 安装 Visual Basic 6.0	4
1.3.3 添加或删除 Visual Basic 6.0 部件.....	5
1.4 Visual Basic 6.0 集成开发环境	5
1.4.1 菜单栏	5
1.4.2 工具栏	11
1.4.3 工具箱	12
1.5 程序工程管理	13
1.5.1 工程类型	13
1.5.2 工程资源	14
1.5.3 工程属性	15
1.6 程序的编辑	16
1.6.1 窗体设计器	16
1.6.2 代码编辑器	17
1.6.3 属性窗口	18
1.7 程序的调试	19
1.7.1 中断调试	20
1.7.2 立即窗口	21
1.7.3 单步调试	21
1.8 程序创建步骤	22
1.8.1 分析程序问题	22
1.8.2 建立程序工程	22

1.8.3	设计用户界面	22
1.8.4	编写源程序	23
1.8.5	调试源程序	25
1.8.6	保存工程	26
1.8.7	编译工程	27
小结		27
习题		29
第 2 章	数据类型与运算	30
2.1	常量	30
2.2	变量	32
2.2.1	声明变量	32
2.2.2	变量的作用范围	33
2.3	基本数据类型	34
2.3.1	数值类型	34
2.3.2	String 数据类型	34
2.3.3	Boolean 数据类型	35
2.3.4	Date 数据类型	35
2.3.5	Object 数据类型	36
2.3.6	Variant 数据类型	36
2.3.7	数据范围	37
2.3.8	数据类型转换	38
2.4	复合数据类型	38
2.4.1	数组	38
2.4.2	枚举类型	41
2.4.3	自定义数据类型	41
2.5	常量、变量命名规则	42
2.5.1	基本规则	43
2.5.2	其他约定	43
2.6	运算符	44
2.6.1	赋值运算符	44
2.6.2	算术运算符	45
2.6.3	比较运算符	45
2.6.4	连接运算符	46
2.6.5	逻辑运算符	46
2.6.6	运算符的优先顺序	47
小结		47
习题		49
第 3 章	程序结构	51
3.1	模块概念	51

3.2	程序过程	52
3.2.1	通用过程	52
3.2.2	函数过程	53
3.2.3	事件过程	54
3.2.4	向过程传递参数的方法及规则	55
3.3	程序流程控制	58
3.3.1	分支结构	58
3.3.2	循环结构	63
3.3.3	嵌套控制结构	67
3.3.4	使用 GoTo 语句	69
3.3.5	使用 DoEvents 语句	69
3.4	程序异常处理	70
3.4.1	出错处理语句	70
3.4.2	通过 Err 对象显示错误信息	72
3.4.3	设计出错处理程序	72
3.4.4	程序运行错误集中处理	74
3.5	常用内置函数	75
3.5.1	数学函数	75
3.5.2	字符串函数	77
3.5.3	转换函数	79
3.5.4	判断函数	79
3.5.5	其他函数	80
3.6	面向对象编程	81
3.6.1	类与对象	81
3.6.2	属性、方法与事件	82
3.6.3	使用对象	83
3.6.4	创建自定义类	86
3.6.5	面向对象程序设计示例	90
3.7	使用 API 函数编程	92
3.7.1	声明 API 函数	92
3.7.2	通过 API 浏览器查看 API 函数	93
3.7.3	API 函数应用举例	95
	小结	96
	习题	99
第 4 章	可视化编程	101
4.1	窗体	101
4.1.1	窗体基本特征	101
4.1.2	窗体中的属性	103
4.1.3	窗体中的方法	106

4.1.4	窗体中的事件	107
4.2	控件	107
4.2.1	内部控件	107
4.2.2	ActiveX 控件	124
4.2.3	控件默认属性	127
4.2.4	控件数组	128
4.3	事件驱动	128
4.3.1	事件概念	128
4.3.2	常见的用户触发事件	129
4.3.3	关联事件的发生顺序	133
	小结	134
	习题	136
第 5 章	用户界面设计	138
5.1	设计菜单	138
5.1.1	标准下拉菜单	138
5.1.2	弹出式菜单	141
5.1.3	运行时修改菜单	142
5.1.4	编写菜单的事件代码	143
5.2	创建工具栏	143
5.3	使用对话框	146
5.3.1	对话框的特点	146
5.3.2	使用对话框函数	146
5.3.3	使用自定义对话框	148
5.3.4	使用公共对话框控件	149
5.4	使用 MDI 窗体	154
5.4.1	MDI 窗体	154
5.4.2	MDI 窗体内的子窗体	155
5.4.3	启动 MDI 应用程序	155
5.4.4	退出 MDI 应用程序	156
5.4.5	优化 MDI 应用程序	157
	小结	157
	习题	158
第 6 章	文件处理与文本输出	160
6.1	使用传统语句或函数处理文件	160
6.1.1	顺序方式访问文件	160
6.1.2	随机方式访问文件	161
6.1.3	二进制方式访问文件	163
6.1.4	文件操作语句与函数	163
6.1.5	App 对象的 Path 属性	165

6.2	使用控件处理文件	165
6.2.1	使用控件搜索文件	165
6.2.2	使用控件编辑文档	167
6.3	使用 FSO 对象模型处理文件	167
6.3.1	文件系统对象编程方法	168
6.3.2	驱动器 (Drive) 对象	170
6.3.3	文件夹 (Folder) 对象	170
6.3.4	文件 (File) 对象	171
6.4	文本输出	174
6.4.1	文本格式化函数	174
6.4.2	文本字体设置	176
6.4.3	文本屏幕显示	177
6.4.4	文本打印输出	179
6.5	示例: 文档编辑器	181
6.5.1	界面设计	182
6.5.2	源程序代码	185
6.5.3	程序编译与运行	192
	小结	193
	习题	194
第 7 章	图形、多媒体与网络编程应用	195
7.1	图形处理	195
7.1.1	坐标系统	195
7.1.2	绘图方法	196
7.1.3	图形的载入与保存	202
7.1.4	实例: 三角函数曲线演示器	203
7.2	多媒体应用	205
7.2.1	Multimedia 控件	205
7.2.2	实例: AVI 动画播放器	208
7.3	网络编程	211
7.3.1	文件传输	211
7.3.2	数据通信	215
7.3.3	实例: 文件下载程序	216
7.3.4	实例: 一对多通信程序	217
7.3.5	实例: 点对点通信程序	221
	小结	224
	习题	225
第 8 章	数据库应用	226
8.1	数据库概述	226
8.1.1	数据库系统	226

8.1.2	Access 关系型数据库	227
8.1.3	SQL 语言	232
8.1.4	数据视图	235
8.2	通过 ADO 对象访问数据库	237
8.2.1	ADO Data 控件	237
8.2.2	ADO 对象模型	239
8.2.3	ADO 对象的编程方法	241
8.2.4	基于 ADO 的数据环境设计器	244
8.3	数据报表设计器	249
8.3.1	报表设计器中的对象结构	250
8.3.2	数据报表设计举例	252
8.4	示例：家庭财务管理程序	253
	小结	267
	习题	269
第 9 章	ActiveX 部件开发	270
9.1	ActiveX 部件的特点	270
9.1.1	ActiveX 部件类型	270
9.1.2	基于部件的分布式应用模型	271
9.2	创建 ActiveX 控件	271
9.2.1	ActiveX 控件创建步骤	271
9.2.2	设定 ActiveX 控件属性	272
9.2.3	ActiveX 控件的关键事件	272
9.2.4	ActiveX 控件的用户事件接口	273
9.2.5	示例：带滚动条的图像控件	273
9.3	创建 ActiveX DLL	280
9.3.1	ActiveX DLL 创建步骤	280
9.3.2	示例：基于 ActiveX DLL 的文本处理器	281
9.4	创建 ActiveX EXE	287
9.4.1	ActiveX EXE 创建步骤	287
9.4.2	示例：基于 ActiveX EXE 的消息公告服务器	288
	小结	293
	习题	294
	参考文献	295

第 1 章 Visual Basic 概述

本章要点:

- Visual Basic 的历史渊源
- Visual Basic 6.0 的特点
- Visual Basic 6.0 的安装
- Visual Basic 6.0 的集成开发环境
- 程序工程管理、编辑与调试
- 程序创建步骤

1.1 Visual Basic 的历史渊源

1.1.1 早期 BASIC

1964 年, 美国计算机科学家 G. Kemeny 和 T. Kurtz 在 Dartmouth 学院创造了 BASIC 语言。这是一种适用于初学者的人机交互式语言, 其名称 BASIC 即是 Beginner's All-purpose Symbolic Instruction Code (初学者通用的符号指令代码) 的简写。

早期 Dartmouth BASIC 是为校园的大学生们创造的, 目的是使大学生能够比较轻松地学习计算机编程, G. Kemeny 和 T. Kurtz 就曾在 Dartmouth 学院使用 BASIC 语言讲授程序设计课程。它非常简单, 仅有几十条语句命令, 只能用于教学, 但学生们对这种入门型语言的印象却极为深刻, 他们认为 BASIC 语言非常容易掌握, 而且在掌握了 BASIC 语言之后, 接着学习 FORTRAN 等其他复杂的高级语言也变得相对简单了。

Dartmouth BASIC 的程序代码是一组编有行号的指令序列, 并须依靠 GOTO 和 GOSUB 等语句进行程序转向与子程序调用; 因此, Dartmouth BASIC 的程序结构很不完善。1976 年, 受当时先进的结构化程序设计思想的影响, S. J. Garland 对 Dartmouth BASIC 进行了结构化改造, 以求 Dartmouth BASIC 有一种比较完善的程序结构。

1983 年, Dartmouth BASIC 被演变成 True BASIC 在市场上销售, 这是一种完全结构化的程序设计语言, 可以在 IBM PC 和 Macintosh 计算机上运行。

1.1.2 Quick BASIC

从 Dartmouth BASIC 到 True BASIC, 代表了 BASIC 的学院式早期形成过程, 它由 BASIC 的创始人所主导。然而, 对于 BASIC 的基于实用原则的市场发展则由 Microsoft 公司所主导。

1975 年, Microsoft 公司的两位创始人——Bill Gates 和 Paul Allen 为 MITS 公司生产的早期个人计算机 Altair 8800 开发了一个 BASIC 解释器, 它成为了 Microsoft 公司的第一件产品。1982 年, 受 IBM 公司委托, Microsoft 公司为个人计算机 IBM PC 开发 DOS 操作系统。在开

发 DOS 的过程中, Microsoft 不仅使 BASIC 解释器成为了 IBM PC 的 ROM-BASIC, 并且通过捆绑在 DOS 系统中加入了 GW-Basic, 于是 BASIC 随 DOS 操作系统一道占领了 PC 市场。

1985 年, Microsoft 公司推出了一种类似于 True BASIC 的完全结构化的 BASIC 语言——QuickBASIC 1.0。QuickBASIC 具有强大的集成开发环境, 并且既包含对 BASIC 代码的解释功能, 也包含对 BASIC 代码的一定的编译功能。因此, 与 True BASIC 追求简约的设计思路不同, QuickBASIC 在设计上更注重语言功能的完整和执行效率的提高, 它成为了那个时代运行速度最快的 BASIC 语言。1991 年, 伴随着 MS-DOS 5.0 操作系统的推出, Microsoft 公司同时推出了 QuickBASIC 的简化版 QBASIC, 并通过与 MS-DOS 捆绑, 而将其作为 DOS 操作系统的成分之一免费提供给了用户。

由于在 BASIC 市场上取得了空前的成功, Microsoft 公司成为了 BASIC 语言在市场发展上的领导者和 BASIC 语言标准的实际制订者。

1.1.3 Visual Basic

1991 年, 随着 Microsoft Windows 操作系统的迅速普及与程序员对于快速图形化应用开发需求的不断增强, Microsoft 公司开发了 Visual Basic 1.0, 其集成了 QuickBASIC 的语法特性与解释编译功能, 并提供了一个可视化的开发环境, 允许程序员在一个所见即所得的图形界面中迅速完成开发任务。为了表明 Visual Basic 是一种能够获得广泛应用的基础程序设计语言, Microsoft 特别将 BASIC 这个名称由完全大写转变成为了大小写混合的 Basic。

1993 年, Microsoft 发布了 Visual Basic 3.0, 使 Visual Basic 具有了支持 ODBC、OLE 的特性。1995 年发布了 Visual Basic 4.0, 为 Visual Basic 引入了类(Class)等面向对象概念。1996 年发布了 Visual Basic 5.0, 使 Visual Basic 具有了基于机器码编译执行的能力。

1998 年发布了 Visual Basic 6.0。Visual Basic 6.0 是传统 Visual Basic 向 Visual Basic.NET 转变前的最后一个版本, 也是传统 Visual Basic 中功能最全、应用最广的一个版本。ADO (ActiveX Data Objects) 技术被引入到了这个版本之中, 并提供了基于 ADO 的数据视图、数据环境设计器与数据报表设计器等; Web 应用系统创建能力有所加强, 新增了 DHTML 应用程序、IIS 应用程序等基于 Web 的应用系统工程; 并提供了功能更加强大的程序创建支持工具, 如: 数据对象向导、数据窗体向导、类生成器实用工具、打包和展开向导等。

为使 Visual Basic 占有更大的市场, 从 1993 年开始, Microsoft 公司还在 Microsoft Office 中提供了一种与 Visual Basic 类似的应用开发环境——Visual Basic for Applications, 它是 Visual Basic 的一个子集, 用于让 Office 用户获得定制应用软件功能的可编程方法。1995 年, 为了满足 Internet 和 IE 浏览器的应用需求, 也为了和 JavaScript 语言展开竞争, Microsoft 公司又推出了脚本语言 VBScript, 在 Windows 平台上, 配合 Microsoft 的 ActiveX、ASP 等技术体系, 程序员们可以使用 VBScript 语言迅速完成基于 Web 的应用系统的创建。

实际上, Microsoft 对于 Visual Basic 的重视, 不仅在于 BASIC 是它们最具历史性的传统产品, 而且也是其战略性发展的需要。在基于 Windows 操作平台的 Microsoft 应用系统发展构架中, Visual Basic 被定义在客户端, 扮演着帮助用户快速创建前端应用程序的角色。因此, 2001 年, 伴随着 .NET 应用平台的推出, Visual Basic .NET 诞生了, 由此 Visual Basic 具有了更加完整、全面的面向对象特性, 具有了和 Visual C++ .NET、Visual C# .NET 一致的基于 .NET 低层支持的内存管理、异常处理机制, 并具有了对 ASP.NET、Web Form 和 Web Services 等新

技术的支持能力。

1.2 Visual Basic 6.0 的特点

由于 Microsoft 对 Visual Basic 高度的重视,使其迅速发展,功能不断强大,不仅继承了 BASIC 简单、易用的特点,同时又具有了可视化、面向对象、部件技术、事件驱动等一系列新的特性,成为了 Windows 环境下最显快捷、便利特性和迄今为止最具影响的快速应用开发工具 (Rapid Application Development, RAD)。

1.2.1 可视化

Visual Basic 中的 Visual 一词即指开发用户界面 (Graphical User Interface, GUI) 的可视化方法。应该说, Visual Basic 之所以成为 Windows 环境下最受欢迎的前端开发工具,其具有的人性化可视开发环境起了很大的作用。

实际上, Visual Basic 已经成为了可视化编程工具的典范,其展现出来的可视性编程风格已经成为了诸多可视化集成开发工具的参照标准。

Visual Basic 的可视化编程特性使编程效率显著提高,例如创建前端用户界面,界面元素的外观和位置就不需要编写大量源代码去描述,而只需要把预先建立好的控件对象用鼠标拖入到设计窗体上即可。

可视化编程带来的好处还体现在原型进化开发模式能得以有效应用上。例如,可以通过 Visual Basic 先建立起应用程序中与用户关系最直接的成分,例如窗体、报表的外观,程序的基本功能等,而其他成分则可随着用户对有待实现的程序系统的需求的不断清晰逐步实现。

1.2.2 面向对象

面向对象编程思想 (OOP) 形成于 20 世纪 60 年代,但直到 20 世纪 80 年代才真正产生了作用,其杰出代表是 C++ 程序设计语言。此后随着 C++ 的广泛应用,面向对象编程语言所具有的优越性得到了很好的验证,于是面向对象编程思想也就被引入到了一些原本基于结构化编程思想的程序设计语言之中,这其中也包括 Visual Basic。

在 1995 年发布的 Visual Basic 4.0 中,类 (Class) 等面向对象的概念被引进来了,由此以来 Visual Basic 也具有了引用类和创建类的能力,并通过 Visual Basic 5.0、Visual Basic 6.0 而不断得以加强。但直到 Visual Basic 6.0, Visual Basic 中的类仍是一个独立的模块文件,没有继承性,只具有功能与数据的封装特性。因此, Visual Basic 6.0 中的面向对象特性是不完整的,这种局面直到 Visual Basic .NET 才有了彻底改变,使 Visual Basic 成为了一种完全面向对象的开发工具。

尽管 Visual Basic 6.0 中的类没有继承性,不能带来基于类的继承的代码重用与功能扩充,但由于类所具有的功能与数据的封装特性能够使系统获得更好的稳定性与可维护性,因此基于类来构建系统也就仍然成为 Visual Basic 最主要的系统实现策略。

1.2.3 部件技术

部件技术建立在面向对象技术基础之上。可将部件看作类的包装,每一个部件包含着许

多相互关联的类，因此具有比单个类模块更加完整的功能。例如，可以将许多与数据操作有关的类包装成一个数据操作部件。

Visual Basic 能够用于部件式开发，例如，可以通过控件来构造用户界面。但 Visual Basic 只能使用符合 Microsoft 部件对象模型标准（Component Object Model, COM）的部件。

值得注意的是，COM 是一种具有二进制标准的对象模型，与类相比，COM 更具通用性，可用于实现跨平台运行的开放式结构。因此，在 Visual Basic 中创建部件可被其自用，也可供其他开发工具（如 Visual C++）使用。当然，Visual Basic 也可以引用由其他开发工具建立的符合 COM 标准的部件。

部件技术为实现软件组装式开发提供了条件，并且也为构造分布式系统提供了便利。可以基于分布式部件对象模型（Distributed COM, DCOM）构造分布式应用系统，例如，可通过 Visual Basic 分别构造客户端部件与服务器部件，然后基于 DCOM 接口标准实现客户端部件与服务器部件的互访，以达到分布式应用的目的。

1.2.4 事件驱动

在面向对象编程中，事件是指对象与外界的感应接口。更具体地说，事件是指系统能够感应到的动作，例如当在窗体或控件上移动鼠标时，系统就能感应到这个事件的发生。

Visual Basic 的每一个窗体和控件都有一个预先定义的事件集。如果其中有一个事件发生，而且，在关联的事件过程中存在代码，则 Visual Basic 调用该代码，这就是所谓的事件驱动。显然，事件驱动程序的执行是依靠系统对事件的感应而触发的。因此，根据事件发生的顺序，可以有效地组织程序的运行。

事件驱动是 Visual Basic 所具有的程序流程基本控制手段。因此，在 Visual Basic 中不仅可以感应到系统预先定义的事件，而且可以在 Visual Basic 中创建自定义事件，以获得更多种类的范围更广的事件触发响应。

1.3 安装 Visual Basic 6.0

1.3.1 Visual Basic 6.0 对环境的要求

为运行 Visual Basic 6.0，必须在计算机上配置相应的硬件系统和软件系统。

具体要求是：

- （1）操作系统：Microsoft Windows NT Server 4.0 或更高版本，Microsoft Windows NT Workstation 4.0 或更高版本，Microsoft Windows 95 或更高版本。
- （2）CPU：486DX/66 MHz 或更高性能的处理器。
- （3）内存：Windows 95/98 下要求 24MB 以上，Windows NT 下要求 32MB 以上。
- （4）磁盘空间：对于学习版，典型安装 48MB，完全安装 80MB；对于专业版，典型安装 48MB，完全安装 80MB；对于企业版，典型安装 128MB，完全安装 147MB。

1.3.2 安装 Visual Basic 6.0

可以按照以下步骤安装 Visual Basic 6.0。

1. 启动安装程序

运行 Visual Basic 6.0 安装光盘上的 Setup.exe, 由此启动安装程序。

2. 输入用户 ID 号

在安装过程中会要求输入用户 ID 号, 可以从软件商提供的资料中找到该 ID 号。

3. 输入姓名、用户单位

在安装过程中会要求用户输入姓名和用户单位。可以在“姓名”输入框中输入用户的姓名, 在“单位”输入框中输入工作单位。单击安装程序界面上的“确定”按钮继续安装。

4. 安装路径的选择

安装程序允许选择安装路径。通过单击安装程序界面上的“更改文件夹”按钮, 可以设置安装路径。

5. 安装方式的选择

有“典型安装”和“自定义安装”两种安装方式。

(1) 典型安装: 在这种安装方式中, Visual Basic 6.0 的所有文件都将安装到硬盘上。

(2) 自定义安装: 按用户的要求安装需要的 Visual Basic 6.0 部件。

Visual Basic 6.0 在经过上述步骤之后, 会进行磁盘空间检查, 然后进行安装。最后, 安装程序会要求重新启动计算机, 并完成安装过程。

1.3.3 添加或删除 Visual Basic 6.0 部件

必要时可随意多次运行安装程序。例如, 可在另一个目录下重新安装 Visual Basic, 也可安装 Visual Basic 的其他部分。

在添加或删除 Visual Basic 部件时, 需要进行以下操作:

(1) 重新启动安装程序。

(2) 在安装程序对话框中选取“自定义”按钮。

(3) 在“自定义”对话框的“选项”列表框中选定要安装的部件 (或撤消选定要删除的部件)。

1.4 Visual Basic 6.0 集成开发环境

Visual Basic 6.0 集成开发环境如图 1-1 所示。其组成包括: 菜单栏、工具栏、工具箱、窗体设计器、属性窗口、工程管理器、代码编辑器, 以及对象浏览器、窗体布局窗口、编程环境设置窗口等。

1.4.1 菜单栏

菜单栏用于组织 Visual Basic 的各项功能, 其置于 Visual Basic 的界面顶端, 有文件、编辑、视图、工程、格式、调试、运行、工具、外接程序、窗口、帮助等菜单栏。

1. “文件”菜单

“文件”菜单涉及建立、打开、添加、移去、保存文件和编译程序等一系列操作 (见图 1-2)。

新建工程: 用于建立一个新的工程, 执行后将显示“新建工程”对话框。